**Operadores Java**

**Símbolos especiais e tipos de operadores**

* Aritméticos
* Unários
* Relacionais
* Lógicos
* Ternário

**Primeiras palavras de programação**

Um programa é um conjunto de instruções lógicas que, quando executadas, produzem algum resultado, logo vemos que há uma base de início, meio e fim que é **recebemos uma informação, trabalhamos a informação e mostramos o exultado**, tudo isso chama-se processamento de dados.

**Operadores de atribuição**

Defini ou sobrescreve o valor de uma variável representado pelo símbolo de igual (=).

* Para declarar um valor em Strings devemos utilizar aspas duplas (“”) envolvendo o texto, no caso de números inteiros não é necessário.
* Valores em com virgulas (decimais) devem ser representados com ponto (.) para separar o inteiro do decimal.
* Booleanos usa-se true (verdadeiro) ou false (falso) sem aspas

Tela de computador com texto preto sobre fundo branco
Símbolo de igual destacado de vermelho entre o nome da variável e o valor

**Operadores Aritméticos**

Utilizados para fazer operações matemáticas complexas ou não.Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

**Concatenação:**

Para unir uma variável a um texto usamos o caractere +, isso se chama concatenação.

Tela de computador com texto preto sobre fundo branco.
Circulo vermelho mostrando a frase, 
1 = "ola, sou " entre aspas,
2 = um caractere + que separa a variável chamada nome
3 = outro caractere + separando o texto entre aspas "e nasci em "
4 = outro caractere + s que separa variável idade

**Operador Unário**Aplicados juntamente com operadores aritméticos, incrementam ou decrementam, invertem valores numéricos e booleanos. Os mais usador são

* (++) adiciona um valor em 1 unidade fazendo uma sequência numérica
* (--) subtrai um valor em 1 unidade decrescendo a sequência numérica
* (!) torna falso o verdadeiro em falso ou o contrário, é o **diferente de**

representação de operador unário

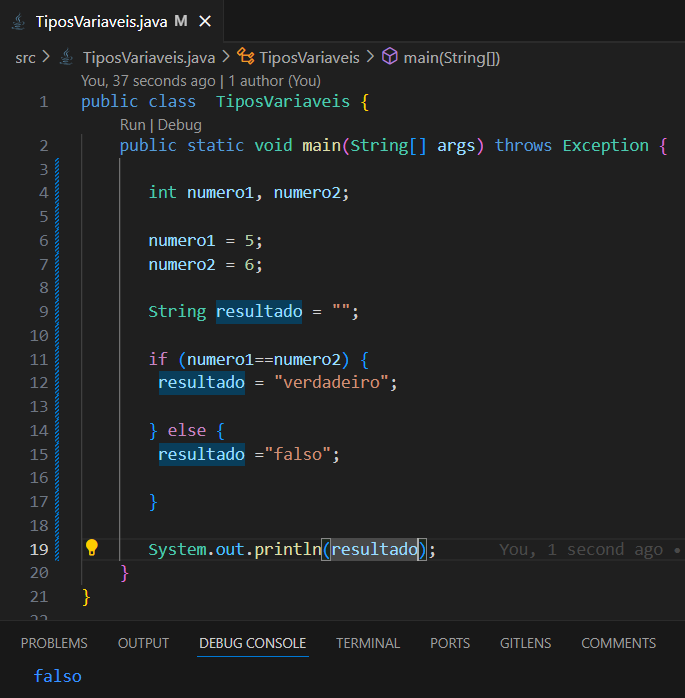

Para retornarmos nosso número para positivo, não podemos utilizar o caractere + pois ele é um operador aritmético. Usamos, então, a lógica matemática de multiplicarmos por -1.

Texto

Descrição gerada automaticamente

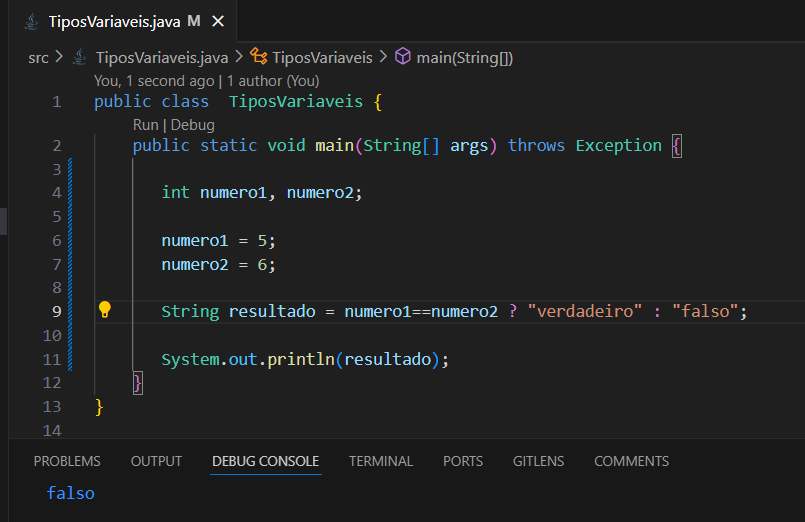
**Operador Ternário**

Este operador deve ser visto como a opção if, ou seja, uma escolha dentre duas saídas, porém escrita em uma única linha. O símbolo utilizado é: **?:**  onde **?** seria a primeira opção e n **:** a segunda opção.



A imagem acima está escrita com **if/else**, porém podemos de uma forma que podemos compreender toda a movimentação lógica da programação.

Vejamos agora com o **operador ternário**.

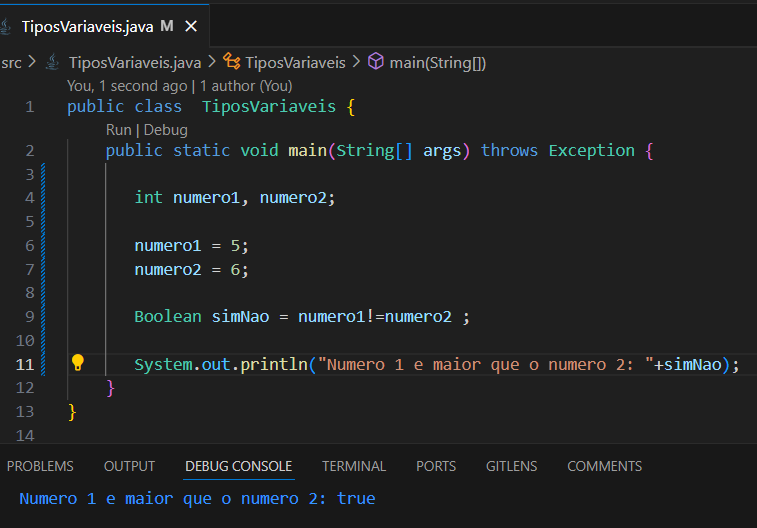


Tudo foi escrito em uma única linha, de forma que podemos entender que, por de trás do operador está sendo executado um if/else, onde **?** seria o **if** e **:** o **else.**

**OBS.:** Operadores ternários serão **sempre usados** em expressões booleanas (true ou false).

**Operadores relacionais**

Faz a comparação entre o elemento da esquerda com o da direita, retornando sempre um booleano.



Quando comparamos dois objetos utilizamos a classe **equals.**

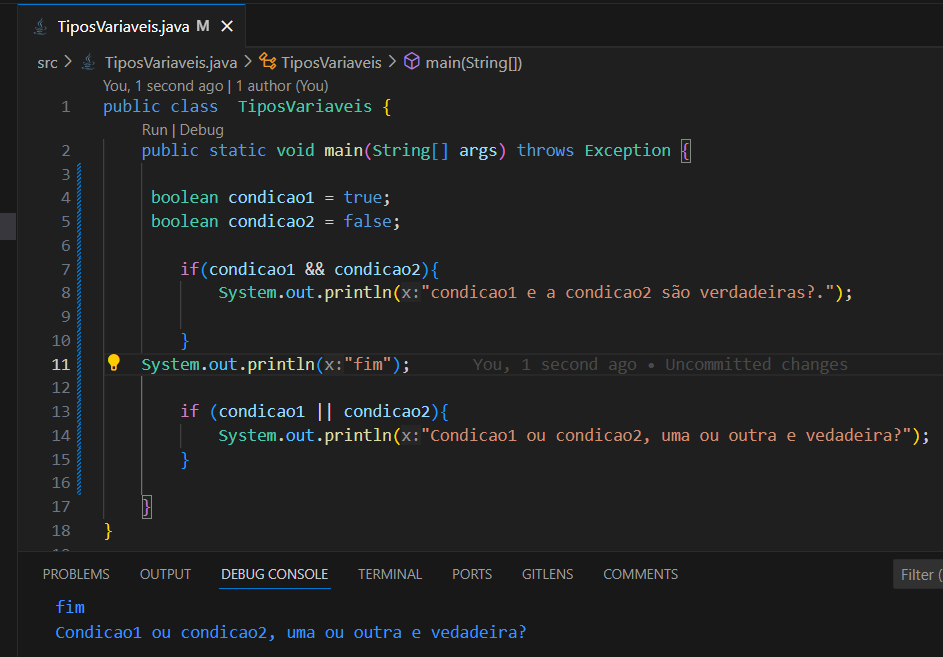
**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

**Operadores lógicos**

Representam ao recurso que nos permite criar expressões lógicas maiores a partir de duas ou mais expressões, através da tabela verdade.

* && operador lógico “E”
* || Operador lógico “OU”



Vemos na imagem que no **primeiro if** a pergunta é: “condicao1 e a condição2 são verdadeiras?”

Como apenas uma das condições são verdadeiras, então foi impresso no console **fim.**

No caso do **segundo if**  a pergunta muda para: “condicao1 ou condição2, uma ou outra é verdadeira?”

Neste caso foi impresso a frase no console, pois uma das duas era verdadeira.